

CLAIMS:

1. A method for calculating a branch target address comprising the steps of:
fetching a branch instruction from a memory, wherein said branch instruction stores an offset of a target address comprising n bits;
calculating n-1 least significant bits of said target address of said branch instruction; and
replacing n-1 least significant bits of said offset of said target address with said n-1 least significant bits of said target address of said branch instruction.
2. The method as recited in claim 1 further comprises the step of:
appending a carry bit to said branch instruction thereby increasing a length of said branch instruction by one bit.
3. The method as recited in claim 2 further comprising the steps of:
calculating a set of upper order bit value combinations of an address of said branch instruction;
storing said branch instruction storing said n-1 least significant bits of said target address in a cache;
retrieving said branch instruction storing said n-1 least significant bits of said target address from said cache; and
selecting a one of said set of upper order bit value combinations of said address of said branch instruction.
4. The method as recited in claim 3 further comprising the step of:
appending said selected one of said set of upper order bit value combinations of said address of said branch instruction with said n-1 least significant bits of said target address to calculate said target address of said branch instruction.

1 5. The method as recited in claim 1, wherein said step of calculating n-1 least
2 significant bits of said target address of said branch instruction comprises the step of:
3 adding a value stored in said n-1 least significant bits of said offset of said
4 target address stored in said branch instruction with a value stored in said n-1 least
5 significant bits of said address of said branch instruction.

1 6. The method as recited in claim 3, wherein said set of upper order bit value
2 combinations of said address of said branch instruction comprises one or more of the
3 following: a value in said upper order bits of said address of said branch instruction
4 incremented by one, said value in said upper order bits of said address of said branch
5 instruction decremented by one and said value in said upper order bits of said address
6 of said branch instruction.

1 7. The method as recited in claim 3, wherein said one of said set of upper order
2 bit value combinations is selected in response to a value in a sign bit and a value in
3 said carry bit in said branch instruction.

1 8. A system, comprising:
2 a memory configured to store instructions;
3 a cache coupled to said memory, wherein said cache is configured to fetch an
4 instruction from said memory; and
5 an encoding logic unit coupled to said cache, wherein said encoding logic unit
6 is configured to encode said fetched instruction, wherein said encoding logic unit is
7 configured to determine if said instruction is a relative branch instruction, wherein
8 said relative branch instruction stores an offset of a target address comprising n bits,
9 wherein if said instruction is said relative branch instruction then said encoding logic
10 unit is configured to calculate n-1 least significant bits of said target address, wherein
11 said encoding logic unit is further configured to replace n-1 least significant bits of
12 said offset of said target address with said n-1 least significant bits of said target
13 address.

1 9. The system as recited in claim 8, wherein said encoding logic unit is further
2 configured to append a carry bit to said relative branch instruction thereby increasing
3 a length of said relative branch instruction by one bit.

1 10. The system as recited in claim 9 further comprises:
2 a fetch unit coupled to said cache, wherein said fetch unit is configured to
3 calculate a set of upper order bit value combinations of an address of said relative
4 branch instruction.

1 11. The system as recited in claim 10, wherein said cache is configured to store
2 said relative branch instruction storing said n-1 least significant bits of said target
3 address.

1 12. The system as recited in claim 11 further comprises:
2 a logic unit coupled to said cache, wherein said logic unit is configured to
3 retrieve said relative branch instruction storing said n-1 least significant bits of said
4 target address from said cache.

1 13. The system as recited in claim 12, wherein said logic unit is further configured
2 to receive said set of upper order bit value combinations of said address of said
3 relative branch instruction from said fetch unit.

4 14. The system as recited in claim 13, wherein said logic unit is further configured
5 to select a one of said set of upper order bit value combinations of said address of said
6 relative branch instruction.

1 15. The system as recited in claim 14, wherein said logic unit is further configured
2 to append said selected one of said set of upper order bit value combinations of said
3 address of said relative branch instruction with said n-1 least significant bits of said
4 target address to calculate said target address.

1 16. The system as recited in claim 8, wherein said encoding logic unit is
2 configured to calculate said n-1 least significant bits of said target address by adding a
3 value stored in said n-1 least significant bits of said offset of said target address stored
4 in said relative branch instruction with a value stored in said n-1 least significant bits
5 of said address of said relative branch instruction.

1 17. The system as recited in claim 10, wherein said set of upper order bit value
2 combinations of said address of said relative branch instruction comprises one or
3 more of the following: a value in said upper order bits of said address of said relative
4 branch instruction incremented by one, said value in said upper order bits of said
5 address of said relative branch instruction decremented by one and said value in said
6 upper order bits of said address of said relative branch instruction.

1 18. The system as recited in claim 14, wherein said one of said set of upper order
2 bit value combinations is selected in response to a value in a sign bit and a value in
3 said carry bit in said branch instruction.

- 1 19. A system, comprising:
2 means for storing instructions;
3 means for fetching an instruction;
4 means for encoding said fetched instruction; and
5 means for determining if said instruction is a relative branch instruction,
6 wherein said relative branch instruction stores an offset of a target address comprising
7 n bits, wherein if said instruction is said relative branch instruction then the system
8 further comprises:
9 means for calculating n-1 least significant bits of said target address;
10 and
11 means for replacing n-1 least significant bits of said offset of said
12 target address with said n-1 least significant bits of said target address.
- 1 20. The system as recited in claim 19 further comprises:
2 means for appending a carry bit to said relative branch instruction thereby
3 increasing a length of said relative branch instruction by one bit.
- 1 21. The system as recited in claim 20 further comprises:
2 means for calculating a set of upper order bit value combinations of an
3 address of said relative branch instruction.
- 1 22. The system as recited in claim 21 further comprises:
2 means for storing said relative branch instruction storing said n-1 least
3 significant bits of said target address.
- 1 23. The system as recited in claim 22 further comprises:
2 means for retrieving said relative branch instruction storing said n-1 least
3 significant bits of said target address.

1 24. The system as recited in claim 23 further comprises:
2 means for receiving said set of upper order bit value combinations of said
3 address of said relative branch instruction.

1 25. The system as recited in claim 24 further comprises:
2 means for selecting a one of said set of upper order bit value combination of
3 said address of said relative branch instruction.

1 26. The system as recited in claim 25 further comprises:
2 means for appending said selected one of said set of upper order bit value
3 combination of said address of said relative branch instruction with said n-1 least
4 significant bits of said target address to calculate said target address.

1 27. The system as recited in claim 19, wherein said n-1 least significant bits of
2 said target address is calculated by adding a value stored in said n-1 least significant
3 bits of said offset of said target address stored in said relative branch instruction with
4 a value stored in said n-1 least significant bits of said address of said relative branch
5 instruction.

1 28. The system as recited in claim 21, wherein said set of upper order bit value
2 combinations of said address of said relative branch instruction comprises one or
3 more of the following: a value in said upper order bits of said address of said relative
4 branch instruction incremented by one, said value in said upper order bits of said
5 address of said relative branch instruction decremented by one and said value in said
6 upper order bits of said address of said relative branch instruction.

- 1 29. The system as recited in claim 25, wherein said one of said set of upper order
- 2 bit value combinations is selected in response to a value in a sign bit and a value in
- 3 said carry bit in said branch instruction.

11/01/2001 10:00 AM RPS20010176US1

1 30. A processor, comprising:
2 a cache configured to fetch an instruction; and
3 an encoding logic unit coupled to said cache configured to encode said fetched
4 instruction, wherein said encoding logic unit is configured to determine if said
5 instruction is a relative branch instruction, wherein said relative branch instruction
6 stores an offset of a target address comprising n bits, wherein if said instruction is
7 said relative branch instruction then said encoding logic unit is configured to calculate
8 n-1 least significant bits of said target address, wherein said encoding logic unit is
9 further configured to replace n-1 least significant bits of said offset of said target
10 address with said n-1 least significant bits of said target address.

1 31. The processor as recited in claim 30, wherein said encoding logic unit is
2 further configured to append a carry bit to said relative branch instruction thereby
3 increasing a length of said relative branch instruction by one bit.

1 32. The processor as recited in claim 31 further comprises:
2 a fetch unit coupled to said cache, wherein said fetch unit is configured to
3 calculate a set of upper order bit value combinations of an address of said relative
4 branch instruction.

1 33. The processor as recited in claim 32, wherein said cache is configured to store
2 said relative branch instruction storing said n-1 least significant bits of said target
3 address.

1 34. The processor as recited in claim 33 further comprises:
2 a logic unit coupled to said cache, wherein said logic unit is configured to
3 retrieve said relative branch instruction storing said n-1 least significant bits of said
4 target address from said cache.

1 35. The processor as recited in claim 34, wherein said logic unit is further
2 configured to receive said set of upper order bit value combinations of said address of
3 said relative branch instruction from said fetch unit.

1 36. The processor as recited in claim 35, wherein said logic unit is further
2 configured to select a one of said set of upper order bit value combinations of said
3 address of said relative branch instruction.

1 37. The processor as recited in claim 36, wherein said logic unit is further
2 configured to append said selected one of said set of upper order bit value
3 combinations of said address of said relative branch instruction with said n-1 least
4 significant bits of said target address to calculate said target address.

1 38. The processor as recited in claim 30, wherein said encoding logic unit is
2 configured to calculate said n-1 least significant bits of said target address by adding a
3 value stored in said n-1 least significant bits of said offset of said target address stored
4 in said relative branch instruction with a value stored in said n-1 least significant bits
5 of said address of said relative branch instruction.

1 39. The processor as recited in claim 32, wherein said set of upper order bit value
2 combinations of said address of said relative branch instruction comprises one or
3 more of the following: a value in said upper order bits of said address of said relative
4 branch instruction incremented by one, said value in said upper order bits of said
5 address of said relative branch instruction decremented by one and said value in said
6 upper order bits of said address of said relative branch instruction.

1 40. The processor as recited in claim 36, wherein said one of said set of upper
2 order bit value combinations is selected in response to a value in a sign bit and a
3 value in said carry bit in said branch instruction.

- 1 41. A processor, comprising:
2 means for fetching an instruction; and
3 means for determining if said instruction is a relative branch instruction,
4 wherein said relative branch instruction stores an offset of a target address comprising
5 n bits, wherein if said instruction is said relative branch instruction then the processor
6 further comprises:
7 means for calculating n-1 least significant bits of said target address;
8 and
9 means for replacing n-1 least significant bits of said offset of said
10 target address with said n-1 least significant bits of said target address.
- 1 42. The processor as recited in claim 41 further comprises:
2 means for appending a carry bit to said relative branch instruction thereby
3 increasing a length of said relative branch instruction by one bit.
- 1 43. The processor as recited in claim 42 further comprises:
2 means for calculating a set of upper order bit value combinations of an
3 address of said relative branch instruction.
- 1 44. The processor as recited in claim 43 further comprises:
2 means for storing said relative branch instruction storing said n-1 least
3 significant bits of said target address.
- 1 45. The processor as recited in claim 44 further comprises:
2 means for retrieving said relative branch instruction storing said n-1 least
3 significant bits of said target address.

1 46. The processor as recited in claim 45 further comprises:
2 means for receiving said set of upper order bit value combinations of said
3 address of said relative branch instruction from said fetch unit.

1 47. The processor as recited in claim 46 further comprises:
2 means for selecting one of said set of upper order bit value combinations of
3 said address of said relative branch instruction.

1 48. The processor as recited in claim 47 further comprises:
2 means for appending said selected one of said set of upper order bit value
3 combinations of said address of said relative branch instruction with said n-1 least
4 significant bits of said target address to calculate said target address.

1 49. The processor as recited in claim 41 further comprises:
2 means for calculating said n-1 least significant bits of said target address by
3 adding a value stored in said n-1 least significant bits of said offset of said target
4 address stored in said relative branch instruction with a value stored in said n-1 least
5 significant bits of said address of said relative branch instruction.

1 50. The processor as recited in claim 43, wherein said set of upper order bit value
2 combinations of said address of said relative branch instruction comprises one or
3 more of the following: a value in said upper order bits of said address of said relative
4 branch instruction incremented by one, said value in said upper order bits of said
5 address of said relative branch instruction decremented by one and said value in said
6 upper order bits of said address of said relative branch instruction.

1 51. The processor as recited in claim 47, wherein said one of said set of upper
2 order bit value combinations is selected in response to a value in a sign bit and a
3 value in said carry bit in said branch instruction.